

CLASS OF 2000

GETTING STARTED IN DEVELOPMENT TODAY IS A TRICKY BUSINESS, ESPECIALLY WHEN YOU HAVEN'T GOT A HOMEBREW PLAYSTATION TO WORK ON. WE CATCH UP WITH THE GENERATION THAT DID...

Way back in the Christmas 2000 edition of **Edge** we gathered together eight upcoming developers whose paths down the game-making road had benefited thanks to their association with Sony's Net Yaroze. This initiative – a £550 piece of tweaked PlayStation 1 hardware along with dev tools and the means to hook it up to PCs/Macs, together with an online support network – is sometimes forgotten in the PlayStation story, but it was a vital tool in the birth of the careers of many console programmers, and its influence is

still being felt today through endeavours such as Microsoft's XNA programme. It wasn't something Sony *needed* to invent, but few who experienced it would say that their lives aren't richer because of it.

Today, we're sitting in an upmarket pie-and-mash joint in Soho to catch up with four of those faces from eight years ago. The others, who aren't able to join us (some no longer even live in the UK), have agreed to share their thoughts via standalone interviews. Together, they have some valuable advice for anyone looking to follow in their footsteps.

How significant was it that Net Yaroze was a PlayStation product?

Robert Swan: When I look back at the Yaroze there were a number of things I wasn't aware of that made it as useful as it was. The Sony brand was attractive, the fact that there was a proper support system was attractive, and then you had a filter, which was the cost of it, that was really valuable. Even today, eight years on, with people who are trying to make games at home, they go to the support websites for PC game-makers, for example, and there's a lot of noise there –

were seeing, and it was on two big screens. I'd never seen the game on a TV as big as that, let alone being seen by so many people. And everything that went with it – we all got tickets to the Sony party that night, and there was a feeling that this was the industry; this was kind of what we're getting into now. I met lots of people through that.

George Bain: I think it was definitely the start of something special, and it was going to lead to a huge community of new programmers in the industry with console programming experience. It was a

"IT'S DIFFICULT TO STAND OUT NOW. YOU'VE GOT TO BE REALLY GOOD OR YOU'RE GOING TO STRUGGLE TO GET INTO THE INDUSTRY"

you know, 90 per cent of the people are saying, 'I've got this great idea for a game – all I need is a programmer, an artist, five designers, all unpaid for two years. By the way, I'm the company owner, and I have nothing to contribute'. So the cost of the Yaroze filtered people out, and what was left was people who were actually going to really try to make games. Sometimes we didn't get very far, but there was the expectation of trying to get support and getting it, and that was very valuable.

What are your fondest memories from the Yaroze period?

Charles Chapman: ECTS was probably the highlight, I think – seeing your game on a major stand. It was a little hidden away but it was something that people

first, and I think it helped generate programmers for PlayStation, not just for Sony but for various other companies, so that in itself was a huge success – I literally created hundreds of programmers who could go on and get a job in the industry. It was a huge benefit.

How important do you think it is for Sony and Microsoft to provide platforms today that will help foster the next generation of game-development talent?

RS: I think it makes good business sense for them to do so. I've been interviewing so many people and there are not enough good programmers out there. Years ago it used to be that people just didn't know what to do





CHARLES CHAPMAN
Then: Founder, Live Media
Now: Founder and tech director, Exient

BOB SHAND
Then: Student
Now: Programmer, Other Ocean Interactive

ROBERT SWAN
Then: Programmer, SCEEE
Now: Lead coder, Nik NaK (Kuju)

GEORGE BAIN
Then: Engineer, SCEEE
Now: Developer relations account manager, SCEEE

NICK PERGUSON
Then: Designer, Ragdoll
Now: Producer, Eidos (*Tomb Raider: Underworld*)

JAMES RUTHERFORD
Then: Programmer, Reflections
Now: Founder, www.creativenucleus.com

TOM MADAMS
Then: Student
Now: Core engineer, LucasArts

JAMES RUSSELL
Then: Engineer, SCEEE
Now: Infrastructure engineer, SCEEE



Photography: Jude Edginton (2000), Kung-fu Kef (2008)

Now they go on courses and I'm becoming pretty critical of a lot of the games courses out there. I'm interviewing a lot of graduates who, if they really cared, would have done extracurricular work, but they haven't. I get a lot of people with a CV that says 'I've done x, y and z', and it looks good, but you get them in and they've done nothing in their spare time. The degree is misrepresenting what is being taught, and the interview is wasted. I cannot hire enough good graduates.

What's the solution?

RS: There are a variety of issues. There are government initiatives and accreditation and skillsets that I'm getting into. I think university courses need to change, and that's something that I'm also slowly getting into. I've started being in touch with various universities, and they're fertile places. A lot of them want to be steered in the right direction – they just

don't know where they're going, so they're picking a direction and the graduates are unemployable.

They are literally unemployable?

RS: They cannot program.

GB: I think the problem is that Net Yaroze and XNA, etc., are very good and very popular, but you can't just have a university degree and say that you've done something on XNA or whatever in order to get a job. A lot of it is down to self-taught programming. I'm still skeptical whether or not a university can teach you what someone can do at home by themselves with the good resources we have on the internet now. There are so many books you can buy now, and there's so much knowledge on the web right now compared to how it was in '97. There were hardly any websites in '97 which had any information about game programming, but nowadays there are so many resources you can use.

When you're looking for new game coders, what sort of demos are you looking for them to show you?

CC: Coders always come with a first-person shooter demo or a graphics demo that shows something you've seen 20 times before. If you're showing a graphics demo it has to be something that's different. Or just show a completed game. Actually, it doesn't need to be a whole game – just something you can play with. We had a guy who'd done some kind of sheepdog simulation – it was just a load of dots on the screen, but he left it with us and for 20 minutes afterwards we just played around with this thing. There probably wasn't much code behind it, but it was fun.

GB: It's very difficult to stand out now, I think. There's so much knowledge now, so many programmers, so much competition, so many teenagers who want to get into the industry. You've got to be really good or you're going to

struggle to get into the industry. You've got to come up with something original to catch someone's eye. I'm glad I got in when I did, put it that way. [Laughter]

If you could go back in time and give your Yaroze-using self any advice, what would it be?

RS: Think about time management. As a student, you leave things until the last minute, and that doesn't cut it in jobs.

But in game development, isn't there a tendency for things to concertina towards the end of the schedule?

RS: They do naturally but they shouldn't. There's nothing in software design that says that's the pattern you have to follow. Individuals shouldn't start with that attitude.

What's your take on XNA?

CC: I think it's a very easy way for people to get involved, but sometimes it's too

easy. From my point of view I'd rather it was based on C++, but the fact that it's out there and picking up a community is better than nothing. Microsoft are looking after themselves a little basing it on C#, because they want to promote that language, but if you were them you would do the same thing. They've got this whole thing that they announced at GDC where they're going to have the big forum and the community and so on, and it's great, it's interesting, but how much finished stuff will get there remains to be seen. I also think that probably a fair amount of stuff on there is actually by professional game developers, anyway: they've got their own idea, they're doing it own their own time, and they'll stick it

expectations because they've got this XNA stuff on their degree, it's not good enough. It's the icing on the cake, but it's not significant enough in itself, and I see that a lot. I would like to see C# as a language spread more, though, because I think it's exceptionally productive for tools, and the game I'm doing at home in my spare time is C#-based because it's that bit more productive. But if I didn't know C++, I would be crippled.

James Rutherford: You're looking at it very much from a programmer's angle, but maybe XNA would be very good for a designer-turned-programmer. The skills aren't quite there if you want someone who's a programmer, but it's good for knocking ideas around.

"SOMETHING LIKE EDUCATION IS WORTHY OF BIG, BOLD STATEMENTS, AND MINE IS THAT IT'S RUBBISH. IT'S NOT DOING WHAT IT SHOULD"

on under a pseudonym. Some of the really good stuff may well turn out to be written by John Carmack or whoever. It will give established people an opportunity to try something with no risks, but that's obviously not what it was designed for. And who knows exactly what it is designed for? Only Microsoft can tell us.

So XNA experience isn't something you'd be looking for on a CV.

RS: I don't know much about XNA but the thing is it shouldn't replace C++ learning. Console development is C++, and if people are coming out with

What's your opinion on the state of further education opportunities for fledgling game-makers nowadays?

RS: I think, with this being an interview, there are a lot of ways you can kind of qualify what you're saying and say, "I don't mean this all the time", but I think something like education today is worthy of big, bold statements, and mine is that it's rubbish. [Laughter] It's really not doing what it should do. I don't care how they present their university – they have to teach this, they have to teach that – I'm trying to hire people and they're rubbish. It does boil down to that.

CC: I'd agree in general that an average

candidate is rubbish – they can't code. There are some exceptions and there are probably some exceptions in terms of the courses as well, so I wouldn't want to slag them all off, but the candidates we're getting from courses, for one reason or another, are not up to it.

RS: The situation is bad at the moment, but I think it's getting better, and I think there's a lot of hope for it to improve more quickly, too. There are more industry people getting involved in these courses. Sometimes they're teaching the courses, but there are various other initiatives, too – Rare, for example, is very closely connected to a certain set of universities,

and there are a couple of guys where I work who are actually connected to quite a few courses, so there's a dialogue going on. However, I think there is more that can be done, and that's why I'm thinking that people like us, if we care enough, we kind of need to do something about it because the rate of change is not going to help me in the next three years.

What do you think needs to be done to give upcoming game-makers the best possible chance to succeed?

RS: I think the first thing is that people need to understand what they're actually signing themselves up for. The idea of

JAMES RUSSELL

How important was Yaroze in terms of you getting a start in the industry?

In one respect it was the sole reason I got into the industry. After graduating from university I had settled in a business software company doing database work, implementing requirements specifications, and drawing up test plans. It was boring and tedious, but I naively thought that's what a real-world job was supposed to be like. I bought the Yaroze to program for fun and because I liked playing around with 3D graphics. One day Sony put up an advertisement on their Yaroze website for a position in their developer support department, and I decided to go for it. It hadn't occurred to me before that I could ever have even tried for a job in the games industry.

In another respect, the experience gained through the Yaroze was valuable for landing the job. Businesses don't want to waste time waiting six months for a raw recruit to get up to speed with their platform, and being able to demonstrate a working Yaroze program was excellent evidence that I knew the platform.

How important do you think it is for console manufacturers to deliver environments to help foster the next generation of game-making talent?

It's a common complaint within the games industry that graduates aren't leaving tertiary education with sufficient real-world skills. I think this will be the case for a long time, because a university's focus is primarily academic while students still view a university education as a stepping-stone to a real-world job. This difference of views certainly applied to me – I learned more about software engineering in my first year of work than I learned in the four years I spent at university studying computer science. That's partly because it wasn't made clear what would be required of me in the real world, partly because of the cost associated with professional software packages, and partly a lack of exposure to real-world development environments. I would expect that most graduates looking to work in the games industry are in the same boat, and so I think it's

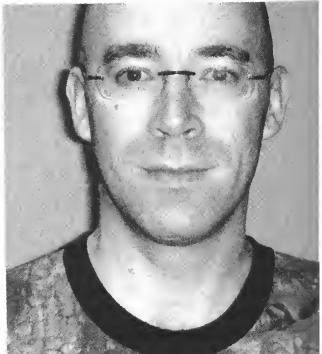
extremely important that the industry attempts to bridge the gap between themselves and academia.

The games industry can assist with all of those obstacles in various ways. Sony's efforts include the Yaroze, PS2 and PS3 Linux, and they've also arranged for selected universities to use PSP development kits. These attempt to address what I think are the three key hurdles. First: knowledge that game development is a real possibility, rather than some lofty and unattainable goal. Second: exposure and access to the development environments. Third: easing the transition from novice to expert by offering good support and documentation.

Students are well-motivated and have the energy to make games. They just lack the tools and need some direction.

If you could go back in time and give your Yaroze-using self any advice, what would it be?

My advice would have been to explore the software engineering side of programming,



rather than just focusing on the fun stuff of how to make graphical effects and optimising routines. A solid background in well-engineered code and working in a team environment is ultimately more valuable than being smart enough to implement a whizzy effect.

TOM MADAMS

How important was Yaroze for you when you were starting out?

Owning a Net Yaroze was what finally made me make the decision to try and break into the industry in the first place. At the time, I couldn't decide whether to choose a career as a games programmer or go out and get a 'proper' job instead. After a couple of weeks programming the Yaroze and being able to show my mates even the most rudimentary games I'd made myself running on an actual PlayStation "just like you've got at home", I was hooked. When it came round to actually applying for work, my Yaroze experience proved invaluable. It wasn't so much that I produced anything particularly impressive with the hardware, more that it showed I had real enthusiasm for the job.

What are your fondest memories of your time using it?

The thing I remember most vividly about the whole Net Yaroze experience was when I got my first demo up and running on the system. It was an incredibly simple program – just a graphic of the joypad not quite in the middle of the screen whose buttons lit up when the corresponding buttons on the controller were pressed. As basic as it was, it seemed almost magical at the time. I could just feel that here was the start of something very cool, something I wanted to spend the rest of my life doing.

How important do you think it is for console manufacturers to deliver environments to help foster the next generation of game-making talent?

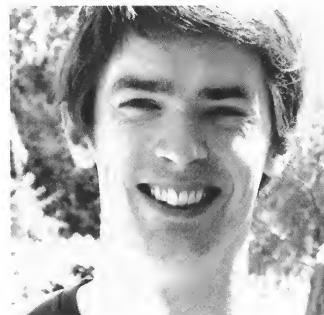
Being a part of Sony's Net Yaroze initiative was an amazing opportunity and is something I wish every budding developer could experience. There were three things that made it so enjoyable for me. Having a fixed platform with a stable framework meant you don't have to worry about any of the boilerplate code that is part and parcel of PC programming – you can just get on with the serious business of making games. It was tremendously liberating, especially in a time before graphics accelerators were commonplace and doing any kind of 3D meant you first had to write your own triangle rasteriser. The kind of community that grows up around initiatives like the Yaroze is a fantastic source of help, advice and most importantly encouragement when you hit problems.

Finally, there's something just that little bit more rewarding about sitting in front of a TV in your front room instead of a desktop when you're playing your latest creation.

I think Microsoft are doing a fine job with their XNA framework – as well as being freely available anyone to try, it's also in use in hundreds of universities. I hope Sony manage to get together a version of their great SPU libraries and package it up in a hobbyists development system for the PS3. I'd love to see what kind of games would be developed with all that processing power in truly indie hands.

What advice would you give your Yaroze-using self?

When you get asked in an interview



"What kind of work would you like to do here?" give them an honest answer, rather than blurting out "I'll do anything!" just because it's a good company and you want a job. Otherwise you'll spend the next two years doing stuff you don't really enjoy. Still, it all worked out nicely in the end.

What's your take on further education opportunities right now?

That's a difficult one. Like a lot of people in the industry, I'm very wary of games development courses. Many of them offer a mixed bag of programming, modelling, animation and design modules, at the end of which a graduate can end up a Jack of all trades, master of none. It's pretty common these days for development teams to number 50 or more, so there's going to be tens of specialists in any one particular area. Someone with a general games development qualification is going to struggle to find a job going up against people with degrees in art, animation or computer science.

As far as programming goes, I'd much rather see more games programming course modules offered as part of conventional computer science degrees. That way, students get the formalised computer science education that employers are after, while still having the opportunity to gain practical experience making video games. Many universities have made steps in this direction with course modules geared towards 'interactive simulations', but there's still a long way to go.

Another problem is that people just don't realise what's involved in making a videogame. Games degrees are becoming increasingly over-subscribed with students who don't really know what they were getting themselves into. The very name of the course gives people who don't investigate actually what's involved something of a false impression. You say to somebody "computer science degree" and it conjures up images of maths, logic and incomprehensible computery things. You say to somebody "videogames degree" and they imagine playing *Halo*, learning the history of *Pac-Man* and maybe doing a paper design for a killer MMORPG. The reality is that, at the core, the courses are very similar, and if you're not ready for that, it will come as quite a shock. I guess that's why there's an above-average drop-out rate for those courses.

somebody who wants to be a game developer and therefore wants to develop in their own time or do a university course, I don't think they have enough information to know what they should be learning and doing. So there's this gap of information. You've got a guy who wants to do something, and this is why the university courses look attractive: "Let's do some Direct3D stuff, some shader stuff, work with Unreal Tournament, UnrealEd" – that sort of thing. At face value that seems pretty relevant but it's actually very, very narrow and not what you'll do in your first job, and I think the same is true of people doing home development – they haven't matured to have enough failures behind them to know you have to keep your scope small, and they don't know how to manage a project from start to finish. I would like there to be some central knowledge that actually tells people what they're signing up for. Once you've done 90 per cent of a game all you've got left to finish is the next 90 per cent, and I think people don't fully appreciate that. And there's a lot of talk by hobbyists and people who want to get into the industry about "I've done this engine" and so on. If you're going to work for a company, they've already got an engine, and if you go and get a job there you're not going to be writing an engine, you're going to be writing a frontend or save routines and so on.

What's your take on PlayStation Network and Xbox Live Arcade in terms of smaller teams getting their work out to the masses?

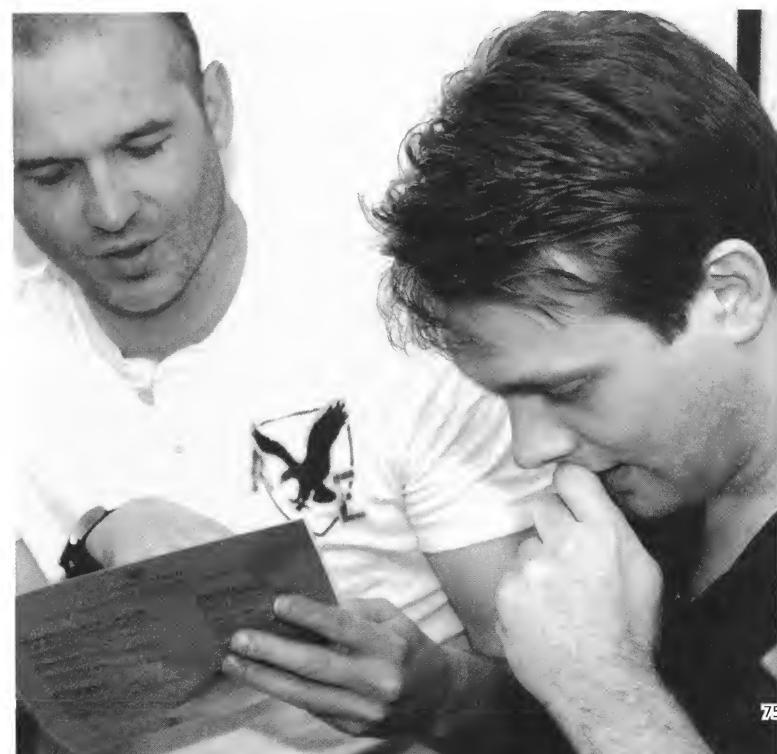
RS: I remember when these things were

announced, everybody was like, "This is going to be a real shot in the arm for the small dev teams – they can do this and they can do that", and that was true for a while, but I think somewhere around a year, a year and a half ago, it became another revenue stream, at which point the bigger companies started sitting up and taking notice, and now the chances of an independent developer getting a slot are very, very low – they're booked up for years. The small-dev-cost-equals-great-little-Xbox-Live-Arcade-release thing, that's becoming less true. The costs are going up.

GB: You still have to be a registered developer to get any content on the PlayStation Network store, but I was saying at a conference to Russian developers in Moscow only recently that Sony is quite open to new independent developers coming on board who want to self-publish their games on PSN. You don't need a thirdparty publisher to have your content released. So we are looking for new content that can be used globally, or content for local markets such as the Russian market or the French market or the German market, etc. To register is quite easy, and we're embracing developers who self-publish.

For example, with *Everyday Shooter*, that was one guy who won the award at the Independent Game Festival. He was based in Toronto, it was just himself doing it, and it was a huge success.

That's a good example of someone doing something that was quite unique on PC; he got spotted by SCEA, and SCEA said, "OK, this is great – do you want to do this for PSN?"



NICK FERGUSON

How important was Yaroze to you?

For me, seeing a Yaroze in action was nothing less than a life-changing 'eureka' moment. I was a pretty hardcore gamer at the time, and it suddenly opened a window into a world where I might be able to do this for a living. I went out the next day, bought *C For Dummies*, and proceeded to neglect my degree for six months. If it wasn't for Yaroze, I would probably be working in the City like all my university friends. So it was a lucky escape, frankly.



What are your fondest memories of your time using it?

Friendly banter on the Yaroze newsgroups. Sitting in my bedroom trying to get something, anything up on screen. The times I got something working on the first attempt were rare and memorable. In fact, they still are!

How important do you think it is for console manufacturers to support?

I think the console factor really draws people into exploring game development. There is little difference between programming for a PC or a 360 using XNA, but the latter certainly sounds cooler.

Anyone over 30 grew up with the Spectrum, C64, Amiga – personal computers disguised as games machines. The technical stuff was always hanging around in the background, LOAD "" or whatever. You almost had to be slightly geeky just to get the games working. If you are 18 now, your first games machine was probably a SNES or Mega Drive – just plug and play – so I think there is a barrier to overcome there.

To be honest, it was the same on Yaroze – I'm sure a lot of the people who bought it never got anywhere.

What is your take on Microsoft's XNA?

I think Microsoft have really stepped up here. XNA is fantastic for homebrew development. It is exactly what I would be looking at if I was starting out today.

I have downloaded the software and experimented with it; it is much easier to get started and better documented than the Yaroze tools were. The ability to share your game with every PC or Xbox Live user is a massive incentive to anyone starting out – much better than hoping your game gets chosen for the OPSM demo disc.

I suppose there is always the underlying concern that it isn't an open platform and Microsoft retain so many rights, but it was exactly the same with Sony and Net Yaroze. In fact, that's just the console business.

What advice would you go back and give your Yaroze-using self?

I wouldn't want to play with the space-time continuum too much – things have worked out pretty well. I think you learn best from your mistakes, which is why Yaroze was so valuable to me: I experienced a lot of failure!

I definitely started out with this notion that good game ideas emerged fully formed, but that's nonsense. You get quality through iteration, trying things – learning what does and doesn't work – and

then throwing out the junk and trying again. That rarely happened in my experience of homebrew development – every line of code is precious, and if you're programming in your spare time you just want to get something done and move on to the next thing. It sounds obvious, but what you realise working in commercial development is that most games become 'fun' right at the end of development after a lot of tinkering – and then you run out of time and the bastards take it off you and put it on the shelf!

What's your opinion on further education opportunities for upcoming game-makers right now?

I've not really been looking closely but there seem to be a lot more options these days. I almost did a Games MSc at Abertay after I graduated, but instead I accepted a job in QA to get my foot in the door. That was the right decision in 1999, but I'm not sure it would be now.

What do you think needs to be done for budding game-makers to be given the best possible chances to succeed?

I think the UK needs to do a better job of nurturing its development community. If I was more cynical, I'd suggest you apply for Canadian citizenship or start learning French instead of C++.

I think the UK scene was healthier when I was starting out. There were more studios, team sizes were smaller and there were always some really diverse, exciting projects in development. It was a good environment to learn in, especially for a generalist like me. I suppose the equivalent now would be working on games for Xbox Live Arcade or PSN.

Consolidation has changed things. These days most studios are a lot bigger and risk-averse, but also more professional and efficient. The big console games need people with deep, expert knowledge – specialists, but who can still work as part of a team. You really need smart people in every discipline to create a triple-A game. I guess the end result is that the competition for the top talent is more intense than ever. Look at *Dare To Be Digital* – the TV companies are even getting involved!

The barrier to entry may be higher, but I don't think that is a bad thing when you look at the quality of games now. When I started out everyone in the office was playing the original *Driver* after hours – now it's *GTA IV*. Thank God for progress!



RS: What interests me is that Xbox Live Arcade had to sell itself at the start – it was: "Come to us, we'll take you on" – but then it reaches a certain point where the priorities shift over to keeping the quantity down but the quality high. And that shift then squeezes out the smaller developers. You can't just publish anything – you can't have another racing game, you can't have another puzzle game, etc. Is that going to happen on PSN? Is it going to happen on WiiWare? I personally think it will, because as it gets successful the numbers increase and you get to be picky and choosy. The smaller guy tends to lose out. But we'll wait and see what happens.

Do you think people working in game development take it more seriously nowadays?

RS: Over the last ten years there's been an improvement in the professionalism of the way companies are run and the way individuals work. It's not been an astounding change, and that's quite disappointing, I think. There's a lot of wisdom outside the games industry that we should be bringing in. You can look at any industry that's been around which has matured. A lot of the companies doing it all badly have just gone and died, and the successful companies have continued. You know, I don't begrudge EA its success; I think they do a lot of things well and they get into a lot of new markets – there's a lot to learn from that – but they still carry a lot of baggage that they really need to leave behind. The quality-of-life thing is getting better, I think – it's on the agenda. I think we should be ashamed of quality-of-life issues. Overtime is a failure and I don't think people stand up and say

that enough. Because we all know it happens, we don't talk about it, which then equates to it being acceptable. And I think we need to draw a line in the sand and say this is just unprofessional. I don't want to get into the discussion of whether it's avoidable or whether it happens – it's just it is unprofessional and we're all to blame. Let's just call a spade a spade. It's like an addiction thing – you can only start overcoming it once you admit it's there. And we don't, really. There's been a lot of soul-searching. It's



ROB SHAND

**How important was Yaroze to you?**

It played a part. Although I think generally it was a small part. When I was younger I was fortunate enough to know what I wanted to do when I grew up. I remember being 13 or so and making the conscious decision that I wanted to make games. I started to learn C/C++ and the Yaroze was another stepping-stone towards my goal of getting a games job. Having some knowledge of the sort of hardware I'd be working on was invaluable, though. Getting the chance to play with the entire machine made me approach things differently and ultimately gave me a greater understanding of the computer architecture.

What are your memories of that time?

My fondest memories related to Yaroze aren't

anything to do with actually coding on it. It was the social side that really made the Yaroze what it was. The worst memory was probably waiting for the thing to arrive, returning from home to find that the courier had tried to deliver it and that it had been taken back to the depot.

Do you think it's important that platform holders put in place initiatives to help develop a new wave of game-makers?

It's of the utmost importance that all console manufacturers foster new talent. There is little point selling a games system without good games to go with it.

What's your take on XNA?

I haven't really looked into it that much, to be

honest. Anything that encourages youths to get involved with development is great. Right now I've got more personal interest in iPhone development – the key difference being that there is a route to market with the iPhone. No matter how slim the chances, there is a chance that you could get your app on the Appstore and make money.

What advice would you give your Yaroze-using self?

For me it would definitely be concentrate on the maths. I could have made my day-to-day life a lot easier had I paid attention instead of playing games. The ability to play games is not equal to the ability to make them. For that you need sound understanding of coding, maths and computer science.

getting better, but overtime is still talked about – you know, it's bad but we still do it, and it's still a sort of badge of honour: "I did a three-night stint", and so on. It's just really, really poor.

JR: Do you not think to a certain extent that if 99 per cent of overtime was got rid of, people would think, 'If I was to stay a couple of hours longer I could make this feature better'? Because it's a creative industry people will always want to make things better than they are. I've never finished a project where I didn't have a huge to-do list of things we could improve, and I'm sure you have as well. I'm given a few weeks at the end

of the project – where you haven't got family commitments or other time commitments – people are naturally going to stay perhaps a bit longer to get things done. And that's probably human nature to a certain extent. If you were a carpenter, and you were making something you really cared about, you'd work into the evenings to get it done.

GB: I think it's difficult now. Looking back, not many people had family commitments but nowadays it's a big thing – a lot of people have grown up and have kids now, they're getting married, etc, so there's lots of commitment that could become challenging when it comes to

It is really difficult when you've got a publisher on the phone at 9pm saying they need a new build, and that's just one of the challenges we face. I think it's a maturity of management issue here. There are many ways you can say that overtime is bad but still actually encourage it, for example saying: "OK, he's working late but he enjoys it, so I'm just going to let that happen" ... We're not as bad as in Japan, where you don't leave until your boss has left, after 14 hours of work, but people pick up on these things because they don't know any better. We should be sending people home. I look back at the times where I've

on overtime, and I don't think that's right. If a company can't survive without enforced overtime or even tacit acceptance of overtime, the company should fold.

Are you talking about the UK?

RS: I'm talking about across the world. I mean, Japan is a great example. A lot of us have this affiliation with Japan – you know, we love the games and the culture and so on – but their work ethic is actually despicable. [Laughter] Maybe we're getting into a cultural thing here but people don't see their families, and we shouldn't be looking at those sort of things and holding them up as examples to follow. The guy who was the designer on *Ico* and *Shadow Of The Colossus* – fantastic games, wonderful, I absolutely adore them – however, I don't want to work for him because it is talked about how he will get rid of people on a whim, how they have to be working their nuts off ten, 12 hours a day. Would I give up those two games to think of people having family lives? No, because I like those games. I just don't think that's a model I'd like to follow.

Are you optimistic about game development's future in the UK?

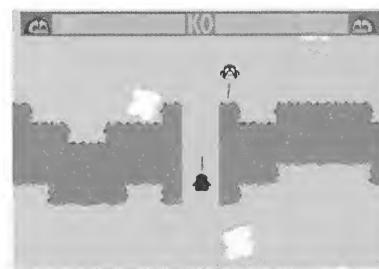
RS: I think the university courses can only improve – they're still very young and I think they will improve, and I think that will help the industry as a whole. There are other issues, broader than just the talent, that need to be covered for the UK industry. There is talent out there and they need to be encouraged to get into the industry in one way or another. It needs to be seen as a good career choice with things like quality of life and salaries and so on – all of those things are important because we need to be competitive.

**"A LOT OF US HAVE AN AFFILIATION WITH JAPAN – WE LOVE THE GAMES AND THE CULTURE – BUT THE WORK ETHIC THERE IS DESPICABLE"**

crunch mode and so on. The industry's never really had to face what it's been facing in the last five years or so, as the whole industry matures.

RS: It's definitely a young industry, though. A lot of graduates are taken on – production people, to a certain extent. People are generally leaving the industry after being in it for a few years, and that's filtering out some of the older people. We've got people with kids and families.

worked late and my life would only have been better if I hadn't done the overtime, and the project would have been better as well. Again, it's a bold statement, but I think it's our job as managers; we know the repercussions of overtime. We should be sending people home. If they want to go home and spend 48 hours doing their own thing that's actually fine, but all the while we're doing this we're relying more and more on a business model that relies



Two of Generation Y's projects: the self-explanatory *Snowball Fight* (left) by James Rutherford, and Bob Shand's *Revolution*. Many Yaroze games were made available to PS1 users via the Official PlayStation Magazine coverdisc

